

anki_vector.status

Robot Status class and exposed properties for Vector's various states.

The `RobotStatus` class in this module exposes properties about the robot status like `is_charging`, `is_being_held`, etc.

Classes

`RobotStatus` ()

A class to expose various status properties of the robot.

`class anki_vector.status.RobotStatus`

A class to expose various status properties of the robot.

`are_motors_moving`

True if Vector is currently moving any of his motors (head, arm or wheels/treads).

```
import anki_vector

with anki_vector.Robot() as robot:
    if robot.status.are_motors_moving:
        print("Vector is moving.")
```

Return type

`bool`

`are_wheels_moving`

True if Vector's wheels/treads are currently moving.

```
import anki_vector

with anki_vector.Robot() as robot:
    if robot.status.are_wheels_moving:
        print("Vector's wheels are moving.")
```

Return type

`bool`

is_animating

True if Vector is currently playing an animation.

```
import anki_vector

with anki_vector.Robot() as robot:
    if robot.status.is_animating:
        print("Vector is animating.")
```

Return type

bool

is_being_held

True if Vector is being held.

`is_being_held` uses the IMU to look for tiny motions that suggest the robot is actively being held in someone's hand.

```
import anki_vector

with anki_vector.Robot() as robot:
    if robot.status.is_being_held:
        print("Vector is being held.")
```

Return type

bool

is_button_pressed

True if Vector's button is pressed.

```
import anki_vector

with anki_vector.Robot() as robot:
    if robot.status.is_button_pressed:
        print("Vector's button was button pressed.")
```

Return type

bool

is_carrying_block

True if Vector is currently carrying a block.

```
import anki_vector

with anki_vector.Robot() as robot:
    if robot.status.is_carrying_block:
        print("Vector is carrying his block.")
```

Return type

bool

is_charging

True if Vector is currently charging.

```
import anki_vector

with anki_vector.Robot() as robot:
    if robot.status.is_charging:
        print("Vector is currently charging.")
```

Return type

bool

is_cliff_detected

True if Vector detected a cliff using any of his four cliff sensors.

```
import anki_vector

with anki_vector.Robot() as robot:
    if robot.status.is_cliff_detected:
        print("Vector has detected a cliff.")
```

Return type

bool

is_docking_to_marker

True if Vector has seen a marker and is actively heading toward it (for example his charger or cube).

```
import anki_vector

with anki_vector.Robot() as robot:
    if robot.status.is_docking_to_marker:
        print("Vector has found a marker and is docking to it.")
```

Return type

bool

is_falling

True if Vector is currently falling.

```
import anki_vector

with anki_vector.Robot() as robot:
    if robot.status.is_falling:
        print("Vector is falling.")
```

Return type

bool

is_head_in_pos

True if Vector's head is in the desired position (False if still trying to move there).

```
import anki_vector

with anki_vector.Robot() as robot:
    if robot.status.is_head_in_pos:
        print("Vector's head is in position.")
```

Return type

bool

is_in_calm_power_mode

True if Vector is in calm power mode. Calm power mode is generally when Vector is sleeping or charging.

```
import anki_vector

with anki_vector.Robot() as robot:
    if robot.status.is_in_calm_power_mode:
        print("Vector is in calm power mode.")
```

Return type

bool

is_lift_in_pos

True if Vector's arm is in the desired position (False if still trying to move it there).

```
import anki_vector

with anki_vector.Robot() as robot:
    if robot.status.is_lift_in_pos:
        print("Vector's arm is in position.")
```

Return type

bool

is_on_charger

True if Vector is currently on the charger.

```
import anki_vector

with anki_vector.Robot() as robot:
    if robot.status.is_on_charger:
        print("Vector is on the charger.")
```

Return type

bool

is_pathing

True if Vector is currently traversing a path.

```
import anki_vector

with anki_vector.Robot() as robot:
    if robot.status.is_pathing:
        print("Vector is traversing a path.")
```

Return type

bool

is_picked_up

True if Vector is currently picked up (in the air).

If `is_being_held` is true, then `is_picked_up` is always True.

`is_picked_up` uses the IMU data to determine if the robot is not on a stable surface with his treads down. If the robot is on its side, `is_picked_up` is True.

```
import anki_vector

with anki_vector.Robot() as robot:
    if robot.status.is_picked_up:
        print("Vector is picked up.")
```

Return type

bool

is_robot_moving

True if Vector is in motion. This includes any of his motors (head, arm, wheels/tracks) and if he is being lifted, carried, or falling.

```
import anki_vector

with anki_vector.Robot() as robot:
    if robot.status.is_robot_moving:
        print("Vector has is in motion.")
```

Return type

bool